

# Meta Data Designer—A Tool for Setting Up Meta Data for a Data Warehouse

Sy Truong (sy.truong@meta-x.com)  
Meta-X Software, Fremont, CA

## INTRODUCTION

One of the main challenges in creating an effective data warehouse is to set up meta data that is consistent with the needs of your data. With proper analysis of existing meta data, it is possible to efficiently generate meta data for the data warehouse.

This paper will examine the importance of meta data for a data warehouse and will present “Meta Data Designer,” a tool facilitating the design of an effective meta data for a data warehouse. It will draw distinctions among different kinds of meta data and demonstrate the importance of meta data in an effective data warehouse. The Meta Data Designer in this example is shown with clinical data, but the concepts could be applied to a range of data warehouses.

## META DATA DEFINED

Meta data is information (or data) about data. Since this definition is so broad, meta data can be interpreted in many ways and may be confusing. One simple example of a meta data for a SAS® data set is the results from a PROC CONTENTS. This provides attributes about the data sets and its variables. Other examples of meta data collected for a data warehouse may pertain to:

- the data structure
- data warehouse table attributes and structures
- mapping from the operational database to the data warehouse
- “currency or timeliness (Delvin 1997, p.55)” of the data: i.e. history of updated time, schedule application run time
- “utilization (Delvin 1997, p.55)”: i.e. tracking use of a particular function, tracking security
- the source data feeding the data warehouse
- the transformation of data as it passes into the data warehouse
- “Meaning (Delvin 1997, p.56)” or an interpretation of the data for a business objective

The previous examples show how meta data can be used to fit the specific needs of a particular warehouse. Since there are so many different kinds of meta data which can be implemented from multiple sources and through many data transformations, it is imperative that meta data be clearly defined and accurately maintained.

Some suggestions you may consider when defining your meta data include the following:

- narrow the scope of each set of meta data to a distinct purpose
- maintain a clear distinction between levels of meta data. For example, the structure of a particular operational data base is stored in a meta data at one level. However, there can also be meta data about the structure of this meta data in level one, which is then stored in another level.
- distinguish between the meta data and the contents of the data (data dictionary issue)
- diagram relationships among types of meta data

An in-depth exploration of the many different kinds of meta data is outside the scope of this paper. The examples in this paper will therefore focus on meta data pertaining to data structures and attributes. This information is most useful for end-users of the data.

An example of a meta data diagram pertaining to the structure and flow of a clinical trials data warehouse is shown below:

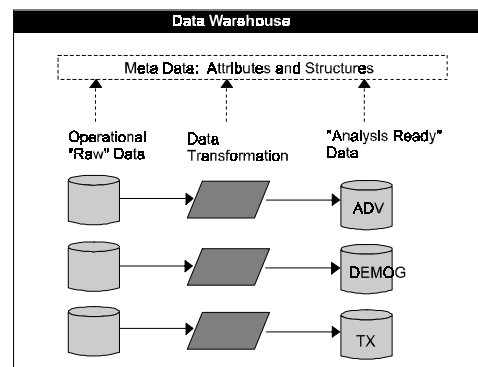


figure 1

The meta data stores attributes and structures of the data during its transformation from the "raw" operational data bases to the "analysis" ready data. Each study within a clinical trial goes through this transformation. Since there are multiple clinical trials conducted simultaneously, the volume of these structures poses challenges for the analyst in charge of the management and development of this data warehouse.

## RATIONALE FOR META DATA

There are several differences between a data warehouse and a classic operational environment which makes meta data even more significant. "In the operational environment, meta data is treated almost as an after thought and is regulated to the same level of importance as documentation. However, meta data in the data warehouse environment takes on a very enhanced role (Inmon 1993, p.133)." "The data warehouse operates under a heuristic, iterative development life cycle (Inmon 1993, p.126)" from multiple sources. This means that the data goes through many transformations and therefore a consistent and accurate meta data is essential. To achieve this accuracy consistently, standards for the meta data also become key.

## METHODOLOGY AND STANDARDS

Since Data warehouse is used among various "information communities," it is natural that it is stored on multiple machines in various formats. This data only becomes useful when it is brought together through the data warehouse by means of data conversions, filtering, summarization, structural changes, etc. All these techniques are applied in a dynamic and iterative fashion. Successful management of the data as it goes through this transformation from the operational environment into the data warehouse is accomplished through an effective implementation of meta data.

"The standardization of meta data can be a tremendous asset to the development of data warehouse and to the analysts that use it (Inmon 1992, p.343)." Standardization makes it easier to catch errors or anomalies within the design. Since there are many data conversions which the meta data captures, standards allow for easier integration among different components. This leads to better reusability and more modularity in design and implementation.

## META DATA DESIGNER

### About Meta Data Designer

Meta Data Designer is a SAS/AF<sup>®</sup> application developed in SAS 6.11 running on UNIX Solaris 2.4. Instead of being designed for a data warehouse administrator, it is geared towards the end user analyst. It does not attempt to set up all types of meta data as mentioned in the meta data definition, but rather focuses on the attributes of the tables in a data warehouse. The Meta Data Designer gathers these attributes from existing data sets in order to create a standard meta data used in the data warehouse. Some of the features include:

- retrieve meta data
- search meta data for key fields
- compare meta data among different tables for similarities
- edit, merge and compile meta data standards from collected meta data

### PROC SQL vs. PROC CONTENTS

In Meta Data Designer, the retrieval of meta data from SAS data sets was first programmed using PROC CONTENTS. An example program is:

```
*** Collect meta data from data set
company ***;
proc contents data = sashelp.company
noprnt
    out = metadata;
run;
```

There are other ways of accessing this meta data, such as:

- PROC PRINT of SASHELP views
- PROC SQL on DICTIONARY tables

Because of efficiency, the PROC SQL on DICTIONARY tables were used for Meta Data Designer. This method was not only efficient in performance during runtime, but was also very suitable for FRAME entry programming with SCL. An example program is:

```
submit sql continue;
    create table sasuser.wh_var as
    select memname, name, label,
        length, format, informat, type
    from dictionary.columns
    where memtype = "DATA";
endsubmit;
```

## Interface

The main window of the Meta Data Designer displays the current data sets which have been collected. It gives options and tools to further add and explore meta data through menus implemented by pmenus and push buttons.

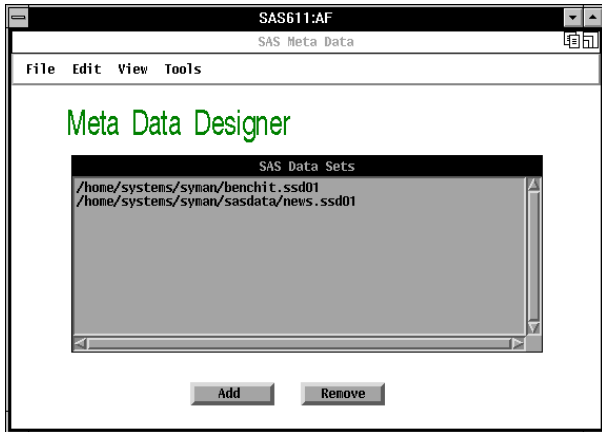


figure 2

The pmenus were implemented using Menu Builder. Menu Builder is a SAS/AF application located at SASHELP.AFTOOLS.PMBMAIN.FRAME. It is a code generator for PROC PMENU which is very handy. However, the documentation for this utility is limited.

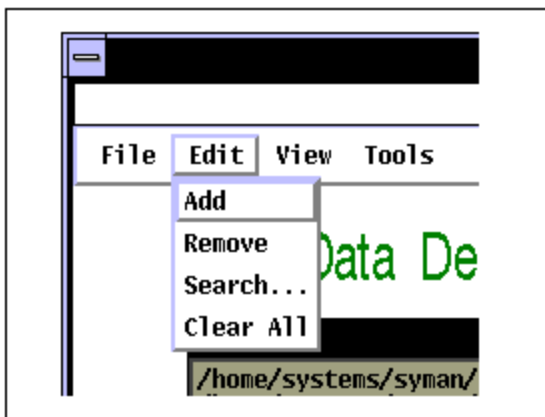


figure 3

Under the Edit menu, the user can add to or remove any meta data set from the dictionary. Once the "Add" menu item is selected, a standard open dialog box appears showing only SAS data sets as shown in figure 4.

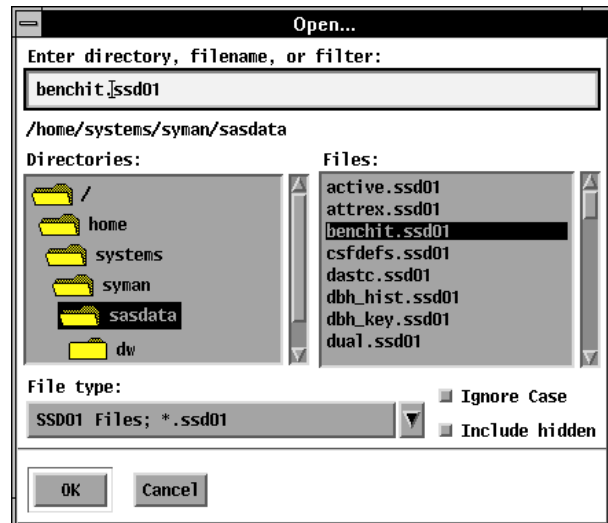


figure 4

Once the data set has been selected, the meta data of this data set is collected and automatically appended to the main dictionary.

To view the meta data details, a browser is supplied as shown in figure 5.

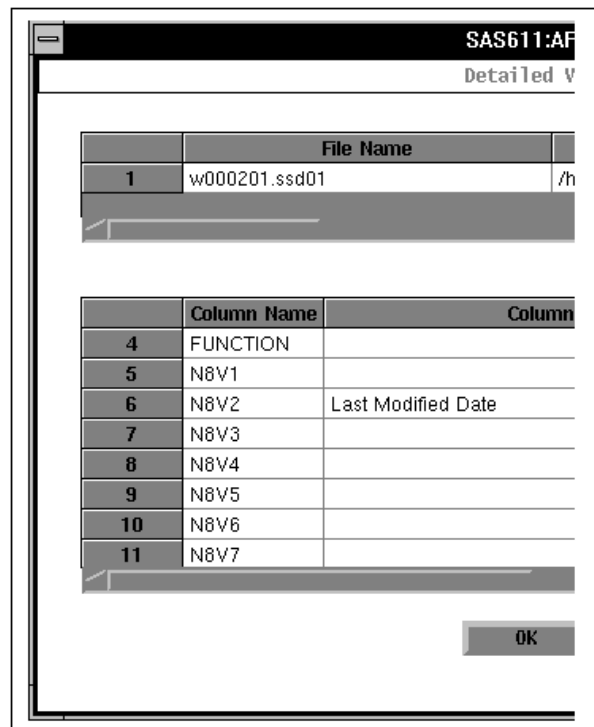


figure 5

As the dictionary grows, a common task is to search for a particular variable name or label. This is accomplished through a dialog box as shown in figure 6.

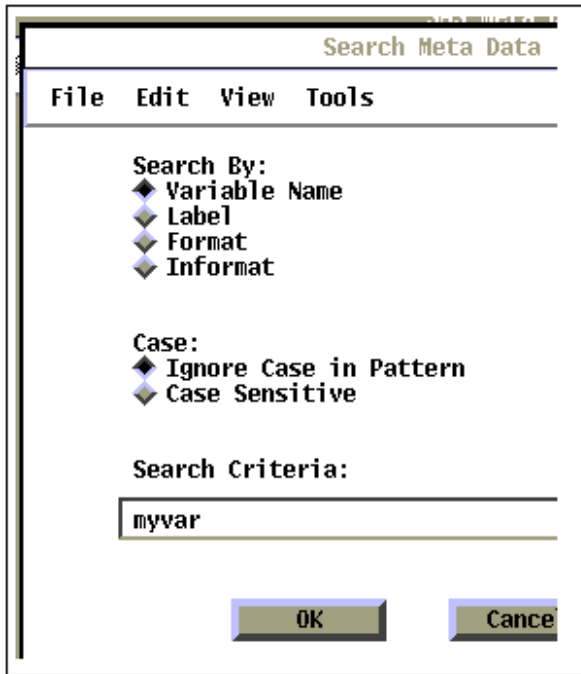


figure 6

The search engine finds any match within a particular field with the like (?) operator. This search is applied to all the data sets contained in its collected dictionary. The example SQL code for this function is:

```
select memname, name, label, format,
       informat, type
  from sasuser.wh_var
 where &category ? "&search";
```

Another common task which can assist in deciding which meta data to use as a standard is to do comparisons among existing meta data. An example for a clinical trials data warehouse is to compare two meta data among treatment files from different studies. If a particular attribute is the same in more than one study, this may suggest choosing this particular attribute as a standard. Figure 7 shows how you can select two data sets within the collected dictionary and compare attributes. This comparison can be applied with any particular attribute such as the name of the variable.

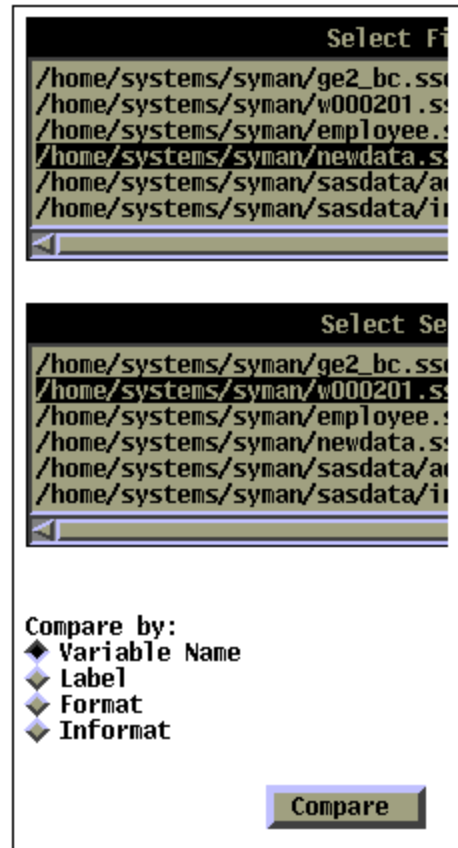


figure 7

After some exploration and comparisons of the existing meta data, the final step is to compile a standard set of attributes deemed appropriate. This is referred to as the final master set. Instead of having to create a new set of attributes from scratch, Meta Data Designer has the ability to capture existing attributes and then populate the final set. The dialog box which does this contains a small browser which also functions as a data set selector as shown in figure 8.

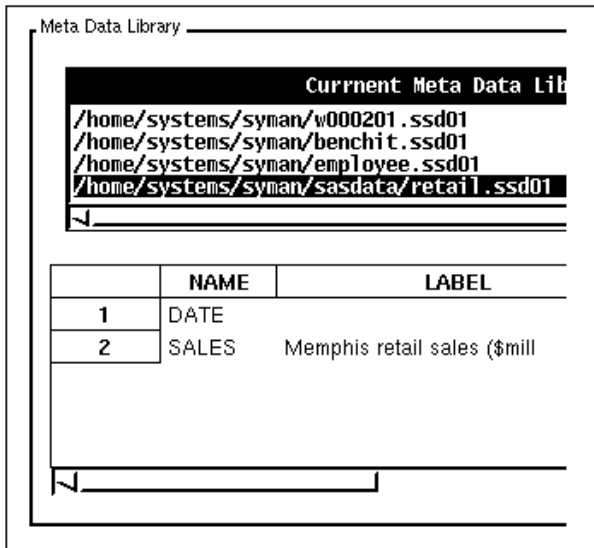


figure 8

Once a particular data set with the associated meta data of interest is selected, you can append this to a final collection of meta data. Figure 9 shows a set of meta data commonly used in a clinical trials data warehouse.

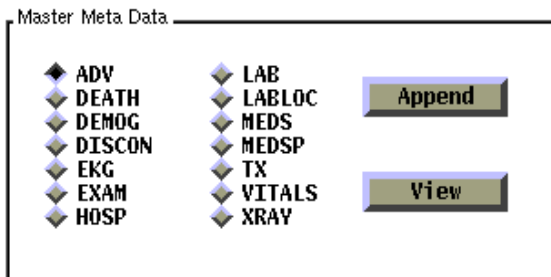


figure 9

There are many other functions which can be added to make the process of designing meta data more effective. There are yet even more tasks involved in maintaining meta data once it is in use. Meta Data Designer however is a small tool which can help you get started in setting up the standard meta data.

## CONCLUSION

The process of creating meta data is both a challenge and an essential component of an effective data warehouse. A clear definition of the type of meta data is important since meta data can be defined in many ways. After a specific scope and definition has been determined, an efficient approach is to design a standardized meta data which closely reflects the existing data structures. The Meta Data Designer utility accomplishes this goal by:

- capturing meta data from existing data
- compiling this meta data into a dictionary
- enabling search and comparison mechanisms
- facilitating the population of standard meta data from existing data

The implementation of AF FRAME gives a user friendly interface for this design process.

## REFERENCES

Delvin, Barry (1997), *Data Warehouse from Architecture to Implementation*, Addison-Wesley Longman, Inc., 55-56

Dilorio, C. Frank & Michael, J. Nancy (1996), *Data About Data: An Introduction to Dictionary Tables*, SUGI 21, 213-219

Grippe, Karen & Dunj, Chen & Braedstreet (1996), *Building a Data Warehouse with SAS Software: A Recipe for Success*, SUGI 21, 686-692

Inmon, W.H. (1993), *Building the Data Warehouse*, A Wiley—QED publication  
John Wiley & Sons, Inc. New York  
1992, 1993 by John Wiley & Sons, Inc., 123-133

Inmon, W.H. & Caplan, Jeffrey H. (1992), *Information Systems Architecture, Development in the 90's*, Wiley QED Publication  
1992 by John Wiley & Sons, Inc., 343

Ross, J. David (1996), *Delving into the Data Warehouse using SAS: Levels of Analysis for a Three-Tier Survey Analysis System*.  
SUGI 21, 508-513

## ACKNOWLEDGEMENTS

SAS, SAS/AF are registered trademarks of SAS Institute Inc. in USA and other countries.  
® indicates USA registration.